

## Labo 2 - Protocoles applicatifs

Eléonore d'Agostino et Karim Ghozlan

05-11-2015

1. **Si une authentification par le serveur est requise, peut-on utiliser un protocole asynchrone? Quelles seraient les restrictions? Peut-on utiliser une transmission différée?**

Oui. Par exemple, avec la classe `HttpURLConnection` d'Android, on peut utiliser un objet `Authenticator` pour définir un nom d'utilisateur et un mot de passe lors des connections HTTP. Par contre, si on n'utilise pas HTTPS, toutes ces valeurs sont envoyées au serveur sans encryption et la connexion n'est donc pas du tout sécurisée.

2. **Lors de l'utilisation de protocoles asynchrones, c'est généralement deux threads différents qui se préoccupent de l'envoi et de la réception. Quels problèmes cela peut-il poser?**

Ceci peut poser des problèmes de concurrence. Par exemple, si on a des ressources partagées entre les deux threads, on peut se retrouver avec des valeurs aberrantes si on tente de modifier une des ressources avec les deux threads en même temps.

3. **Lorsque l'on implémente l'écriture différée, il arrive que l'on ait soudain plusieurs transmissions en attente qui deviennent possibles simultanément. Comment implémenter proprement cette situation:**

- (a) **Effectuer une connexion par transmission différée**

-

- (b) **Multiplexer toutes les connexions vers un même serveur en une seule connexion de transport. Dans ce dernier cas, comment implémenter le protocole applicatif, quels avantages peut-on espérer de ce multiplexage, et surtout, comment doit-on planifier les réponses du serveur lorsque ces dernières s'avèrent nécessaires?**

-

- (c) **Comparer les deux techniques et discuter des avantages et inconvénients respectifs**

-

4. **Quel inconvénient y a-t-il à utiliser une infrastructure comme JSON n'offrant aucun service de validation par rapport à une infrastructure comme SOAP ou XML-RPC offrant ces possibilités? Y a-t-il en revanche des avantages que vous pouvez citer?**

Vu qu'il n'a pas de service de validation, on ne peut pas vérifier que le contenu ou la structure du JSON est correct avant que le serveur l'ait reçu et tente d'en utiliser le contenu, et donc on ne peut pas prédire comment le serveur réagira. Alors qu'une infrastructure ayant un service de validation pourra tenter de valider le message reçu, et s'il n'est pas valide, pourra agir en conséquence.

Par contre, JSON a l'avantage d'être plus léger à envoyer.

5. **Quel gain peut-on espérer en moyenne sur des fichiers texte en utilisant de la compression?**

-